INTERNATIONAL STANDARD

ISO/IEC 21972

First edition
2020-01

Information technology — Upper level ontology for smart city indicators

Reference number
ISO/IEC 21972:2020(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Fax: +41 22 749 09 47
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see http://patents.iec.ch).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

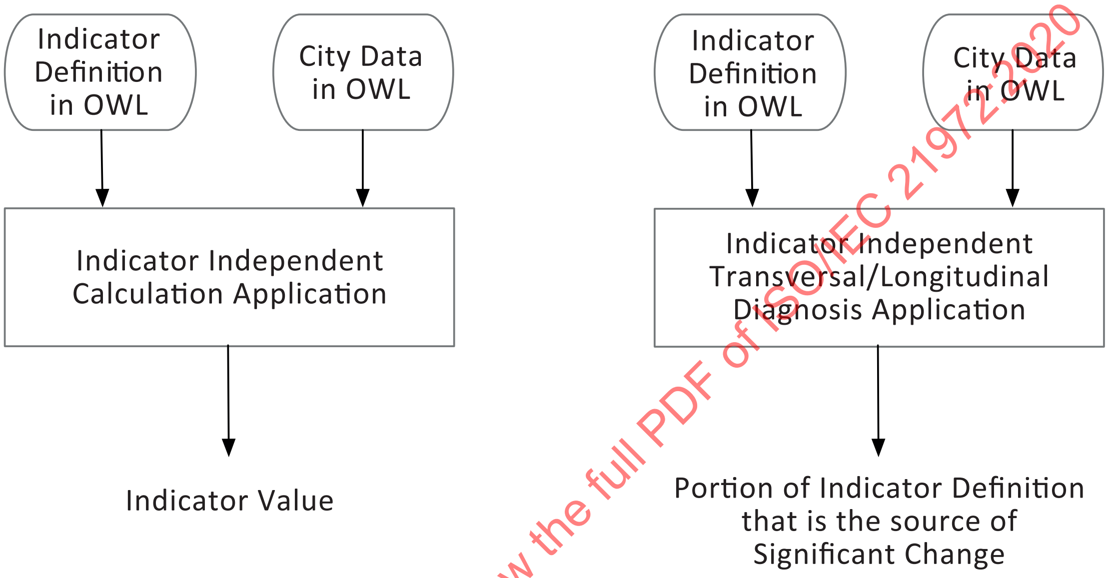
For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

To paraphrase Lord Kelvin, you cannot manage what you cannot measure[11]. For cities to be smart, their decisions need to be based on precisely defined and accurate metrics. For smart city information and communication technology to be used to aid cities in making smart decisions, then the digital data models they use need to precisely and accurately reflect what they represent of the city and how it is measured. This document specifies a data model that can be used to represent city indicator definitions. The data model is defined using the Semantic Web OWL 2 Web Ontology Language (OWL). Figure 1 depicts two intended uses of this document.



a) **Definition-based Calculation of Indicator from Base Data**

b) **Definition-based Diagnosis of Indicator-based City Performance**

Figure 1 — Possible uses of this document

Figure 1 a) depicts the indicator definition being used to automate the computation of an indicator value. In this case, an indicator definition plus city data is input into the indicator independent calculation application, which uses the definition to select subsets of city data, to compute the indicator. This approach makes it possible to create an indicator calculation application that is not programmed for a specific set of indicators. Figure 1 b) depicts a diagnosis system that uses the definition of an indicator as a basis for determining the root cause of transversal or longitudinal deviations in an indicator's value over place or time. A diagnosis system must understand what data was selected and how it was combined in order to determine the sources of change. In the remainder of this Introduction, the motivation for and the purpose of this document are elaborated.

Cities are moving towards policy-making based on data[33]. Yet it has been recognized by urban researchers, city professionals and political leaders that city level data is both incomplete and inconsistent. In 2007, it was recognized that "there are thousands of different sets of city (or urban) indicators and hundreds of agencies compiling and reviewing them. Most cities already have some degree of performance measurement in place. However, these indicators are usually not standardized, consistent or comparable (over time or across cities), nor do they have sufficient endorsement to be used as ongoing benchmarks."[27]

In response, ISO 37120 was developed to provide a set of indicators, across 17 themes, to measure city performance. These indicators spanned areas such as education, finance, shelter, transportation and environment.

Indicator definitions are people oriented; they are provided in natural language, e.g., English, and not in a more formal, possibly computer readable language. The reader of the definition imposes their own interpretation of the definition based on their understanding of the language and the environment in which they live (e.g., how their own city may define some terms).

Consider the definition of a student/teacher ratio as provided in Reference [21]: "Student/teacher ratio". This has been expanded[34] to: "Student/teacher ratio", where the numerator is "Number of Students", and the denominator is "Number of Teachers". One problem is whether "student" refers to full time students, or part time students. Are they regular students or special needs students? Do they include kindergarten students or not? It is also difficult to compare an indicator for a single city across time if the definition of student changes. For example, today the educational system includes students with special needs, but 30 years ago they may not have been enrolled. Without a more precise definition of terms, it makes it difficult to compare an indicator across cities where each city interprets what a student is differently, or against itself where definitions change.

Obviously, the definition and documentation of indicators can be expanded, as has been done in ISO 37120:2018, 6.4.2.

The definition of student/teacher ratio clearly addresses some of the issues raised above. Nevertheless, there is always a disconnect between the actual value of a city's indicator and the data sources and processes used to measure it; while the indicator's value is recorded in a machine-readable form (e.g., database or semantic web), the sources and measurement processes are buried in datasets and documents that are inaccessible or only human readable. In the end, all that is left is a record of indicator values without an understanding of what they actually measure and how they were measured.

The purpose of this document is to support the precise and unambiguous specification of indicator definitions using the technology of ontologies[24][25] as implemented in the semantic Web[13]. By doing so, it:

— enables the computer representation of precise definitions thereby reducing the ambiguity of interpretation;

— takes indicators out of the realm of humans and into the realm of computers where the world of big data, open source software, mobile apps, etc., can be applied to analyze and interpret the data;

— achieves semantic interoperability, namely the ability to access, understand, merge and use indicator data available from datasets spread across the semantic web;

— enables the publishing of indicator definitions, indicator values and their supporting data using semantic Web and ontology standards;

— enables the development of indicator independent indicator calculation applications; and

— enables the automated detection of indicator data inconsistency, and the root causes of variations.

Without a clear semantics for indicator definitions, it is not possible to perform consistency analysis. Without determining consistency, the ability to validate any comparisons based on indicators is lacking.

In this document, the indicator upper level ontology (IULO) is introduced. The IULO provides the concepts and properties for representing the basic structure of the definitions of indicators (see Clause 6). It does not provide concepts for representing theme specific concepts, such as education, finance, shelter, etc.

The IULO has been devised to communicate the meaning of data. It does not attempt to provide concepts to describe the metadata of indicators, for example, validity and provenance of data.

The IULO does not replace existing data models where they exist, but by mapping from a local model to the IULO, semantic interoperability of data can be achieved.

The IULO has been devised to represent any aggregate level of indicator, whether it is for neighbourhoods, villages, cities, states/provinces and/or countries.

The IULO has been devised to represent any indicator, and is not restricted to indicator standards, such as ISO 37120, which is normative to this document.

This document is aimed at organizations that define indicators, the information and communications technology (ICT) organizations that provide services to cities, states and countries, and manage the resulting data, as well as ICT and open data developers.

This document is based on work developed in the Enterprise Integration Laboratory of the University of Toronto[18][20][21].

Information technology — Upper level ontology for smart city indicators

1 Scope

This document establishes general principles and gives guidelines for an indicator upper level ontology (IULO) for smart cities that enables the representation of indicator definitions and the data used to derive them. It includes:

— concepts (e.g., indicator, population, cardinality); and

— properties that relate concepts (e.g., cardinality_of, parameter_of_var).

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced documents (including any amendments) applies.

ISO 37120:2018, *Sustainable development of communities — Indicators for city services and quality of life*

"Time Ontology in OWL, W3C Recommendation 19 October 2017". Accessed at https://www.w3.org/TR/owl-time/

ISO 4217, *Codes for the representation of currencies*

ISO 80000 (all parts), *Quantities and units*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 37120 and the following apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at http://www.electropedia.org/

3.1
cardinality
number of elements in a set

[SOURCE: ISO/IEC 11179-3:2013, 3.2.13]

3.2
description logic
DL
family of formal knowledge representation languages that are more expressive than propositional logic but less expressive than first-order logic

3.3
manchester syntax
user-friendly compact syntax for OWL 2 ontologies

Note 1 to entry: The syntax is frame-based (as opposed to the axiom-based other syntaxes for OWL 2) where a collection of information about a class or property is given in one large syntactic construct, instead of being divided into a number of atomic chunks [as in most *description logics* (3.2)] or even being divided into even more triples (as when writing OWL as RDF graphs [RDF Concepts]).

[SOURCE: https://www.w3.org/TR/owl2-manchester-syntax/]

3.4
measure
value of the measurement (via the numerical_value property) which is linked to the both Quantity and Unit_of_measure

3.5
namespace
collection of names, identified by a URI reference, that are used in XML documents as element names and attribute names

3.6
ontology
formal representation of phenomena of a universe of discourse with an underlying vocabulary including definitions and axioms that make the intended meaning explicit and describe phenomena and their interrelationships

[SOURCE: ISO 19101-1:2014, 4.1.26]

3.7
OWL 2 Web Ontology Language
ontology language for the *semantic Web* (3.8) with formally defined meaning

Note 1 to entry: OWL 2 ontologies provide classes, properties, individuals, and data values and are stored as Semantic Web documents.

[SOURCE: https://www.w3.org/TR/owl2-overview/]

3.8
semantic Web
W3C's vision of the Web of linked data

Note 1 to entry: Semantic Web technologies enable people to create data stores on the Web, build vocabularies, and write rules for handling data.

[SOURCE: https://www.w3.org/standards/semanticweb/]

3.9
unit_of_measure
definite magnitude of a quantity, defined and adopted by convention and/or by law

4 Symbols and abbreviated terms

OWL Ontology Web Language

RDF Resource Description Framework

In the figures, arrows with a closed arrowhead denote the "rdfs:subClassOf" relation. Arrows with an open arrowhead denote an attribute relation and have the name of the attribute attached. Where a prefix (denoted by a ":") appears in front of a class or attribute name, the prefix denotes the namespace, the class or attribute originate. Colours of classes (boxes) in the diagrams are used to enhance readability, to distinguish the namespaces from which they originate. They correspond to the namespaces that concepts and properties are drawn from. Each namespace prefix is included in the boxes and redundant with the colour. A subset of concepts is incorporated into this document as OM is not a standard, enhance not normative. Classes and attributes without a prefix are defined in this document.

The following namespace prefixes are used in this document:

— ex: signifies an example for which an IRI does not exist

— gcie: http://ontology.eil.utoronto.ca/GCI/Education/GCI-Education.owl#

— geo: http://www.geonames.org/

— gis: http://www.opengis.net/ont/geosparql#

— om: http://www.wurvoc.org/vocabularies/om-1.8/

— time: http://www.w3.org/2006/time#

— owl: http://www.w3.org/2002/07/owl#

— rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns#

— rdfs: http://www.w3.org/2000/01/rdf-schema#

— sc: http://schema.org/

— xsd: http://www.w3.org/2001/XMLSchema#

5 Basic indicator ontology pattern

Indicator definitions conform to a basic ontology pattern. Figure 2 depicts the pattern (only a portion is shown). For example, an ISO 37120 indicator (usually) has associated a unit of measure that is the ratio of two populations, the year it was measured, and the city it is for. It is composed of a numerator and a denominator which are both quantities. Each is a quantity that is a measure of the size of a population. (Cardinality can be replaced with other measures of a population, such as the mean of a specific property such as age.) The population members are defined, in this example, by another class and city. Concept names without prefixes are defined in this document. Concept names with prefixes are imported into this document. The prefix "time" refers to OWL-Time, and "sc" to Schema.org.

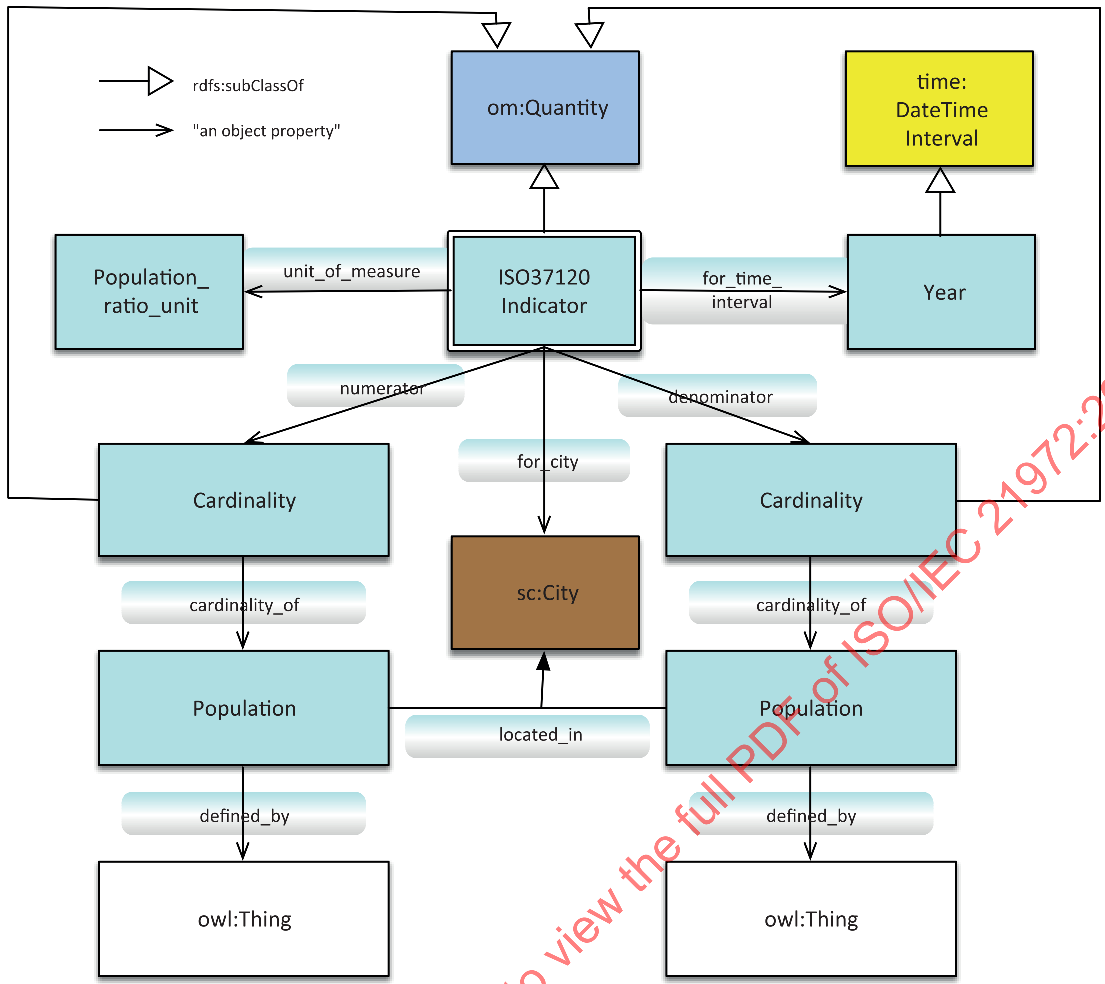


Figure 2 — Indicator ontology pattern

The following clauses describe and define the classes and properties for representing indicator definitions. Formal specifications are provided in the OWL 2 Web Ontology Language[8][9] using a subset of the manchester syntax.

6 Time

6.1 General

Indicator data in a smart city need to be understood in the context of the time at which the data were generated and/or published. It is important to understand not just at what time something occurred, but whether something occurred before, after or during some other event. To answer these questions, a much richer understanding of time that supports reasoning about time points, time intervals and the relationships among them is needed. In summary, the time ontology needs to be able to support the answering of questions such as:

— At what time did some event or measurement occur?

— What was the duration of the event?

— Did the event occur before, after or during some other event?

Many time ontologies have been developed. "Time Ontology in OWL W3C Recommendation 19 October 2017" shall be used in the context of this document.

6.2 Core classes and properties

Fundamental to any conceptual model is the time at which things occur. For example, questions can arise regarding the temporal relationship among measurements, i.e. not just at what time something was measured, but whether it was measured before, after or during some event. For example, was "Total Employment" of New Orleans determined before or after Hurricane Katrina? Or did Katrina take place during the interval that the indicator was determined? To answer these questions, a notion of time that supports reasoning about time points, time intervals and the relationships amongst them is needed. The following summarizes a subset of classes and relationships in OWL-Time.

There are three top level classes:

— **TemporalEntity**: it specifies the two types of time: Instant and Interval.

— **DateTimeDescription**: a specification of a date and time using a year, month, day, hour, etc. set of properties.

— **DurationDescription**: a class that specifies a duration as any combination of years, weeks, days, hours, minutes, and seconds. Equivalent to ISO 19108 'TM_PeriodDuration'.

A TemporalEntity has 3 sub-classes:

— **Instant**: it represents a point in time. Equivalent to ISO 19108 'TM_Instant'.

— **Interval**: it represents a period of time with a beginning and an end. Equivalent to ISO 19108 'TM_Period'. If a DurationDescription is provided, then the difference between the beginning and end of the Interval should be equal to the DurationDescription.

— **ProperInterval:** it is an Interval where the beginning time is less than the end time.

A **TemporalEntity** has a beginning Instant, an ending Instant and a duration, which are denoted by the following properties:

— **hasBeginning**: links a TemporalEntity (domain) to an Instant (range) where the latter denotes the beginning of the TemporalEntity. Equivalent to ISO 19108 'Beginning'.

— **hasEnd:** links a TemporalEntity (domain) to an Instant (range) where the latter denotes the end of the TemporalEntity. Equivalent to ISO 19108 'Ending'.

— **hasDurationDescription**: links a TemporalEntity (domain) to an Interval (range) where the latter denotes the duration of the DurationDescription.

NOTE Properties in RDF are uni-directional, linking a subject to an object. The domain of a property restricts what the subject can be, and the range restricts what the object can be.

Finally, there is a set of properties that relate ProperInterval's, including intervalOverlaps, intervalAfter, intervalContains, etc. Since both OWL-Time and ISO 19108 are based on Allen's temporal[12], each temporal relation in OWL-Time has an equivalent in ISO 19108[7].

6.3 Graphical depiction

The directed graph in Figure 3 depicts the core classes that comprise OWL-Time.

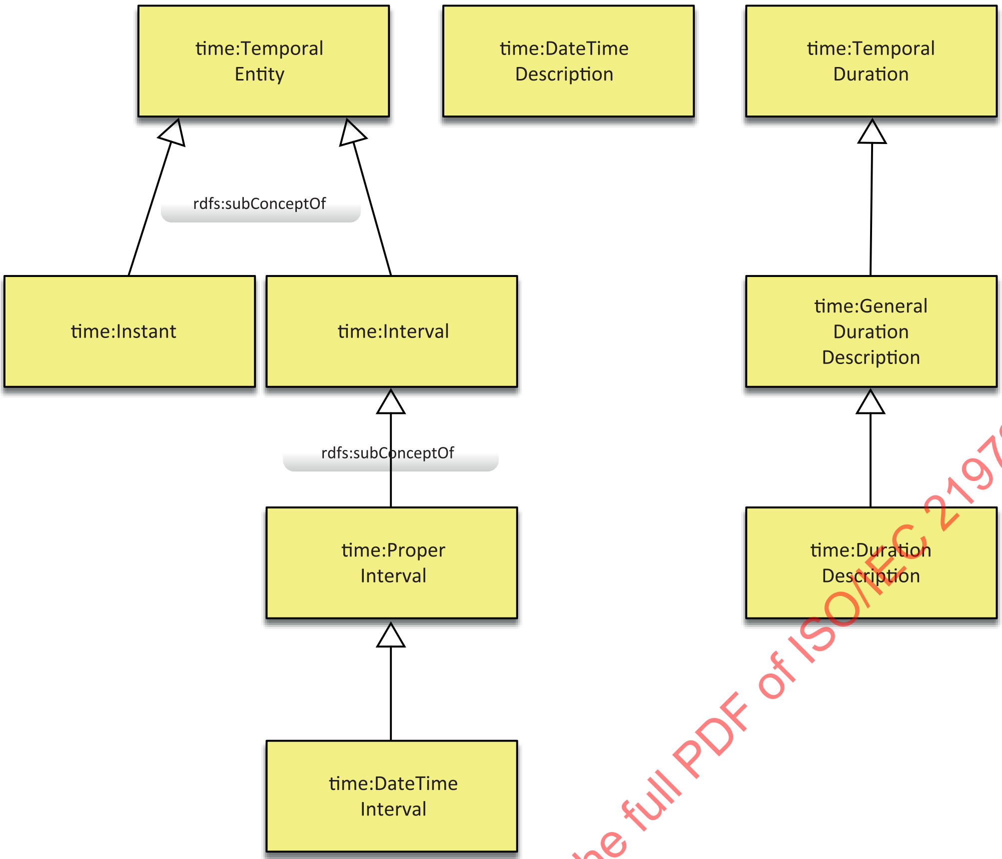


Figure 3 — Time concepts

Figure 4 depicts the relationships that `TemporalEntity` has with the other classes.

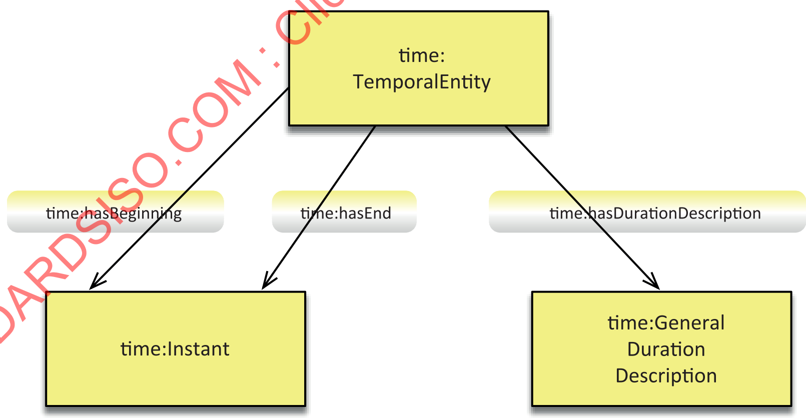


Figure 4 — TemporalEntity relationships

7 Quantities and units of measure

7.1 General

The representation of measurement concepts is based on the OM measurement ontology[31]. The purpose of a measurement ontology is to provide the underlying semantics of a number, such as what is being measured and the unit of measurement. The importance of grounding an indicator in a measurement ontology is to assure that the numbers are comparable, not that they are measuring the same thing, but the actual measures are of the same type, e.g., the counts of the student and teacher populations, that comprise the ratio of student and teacher population sizes, are of the same scale, for example, thousands vs millions. In the diagrams within this clause, classes and properties drawn from the OM ontology are prefixed with "om". The prefix is removed in the formal specification as the identified subset of OM classes and properties are incorporated in the document.

7.2 Core classes and properties

The top row of Figure 5 depicts the basic classes of the measurement ontology. There are three main classes:

— a Quantity that denotes what is being measured, e.g., diameter of a ball, and links to the actual thing being measured via the phenomenon property, and the amount of the quantity via the value property that links to a Measure;

— a Unit_of_measure, which is used as a standard for measurement of the same quantity, where any other value of the quantity can be expressed as a simple multiple of the unit of measure. For example, length is a quantity; the metre is a unit of length that represents a definite predetermined length. When we say 10 metre (or 10 m), we actually mean 10 times the definite predetermined length called 'metre'."[31]

— a Measure that denotes the value of the measurement (via the numerical_value property) which is linked to the both Quantity and Unit_of_measure.

For example, Student Teacher Ratio is a subclass of Quantity that has a value that is a subclass of Measure whose units are a 'population ratio unit' that is an instance of Unit_of_measure. The actual value measured is a property of the Measure subclass 'Student teacher ratio measure'.

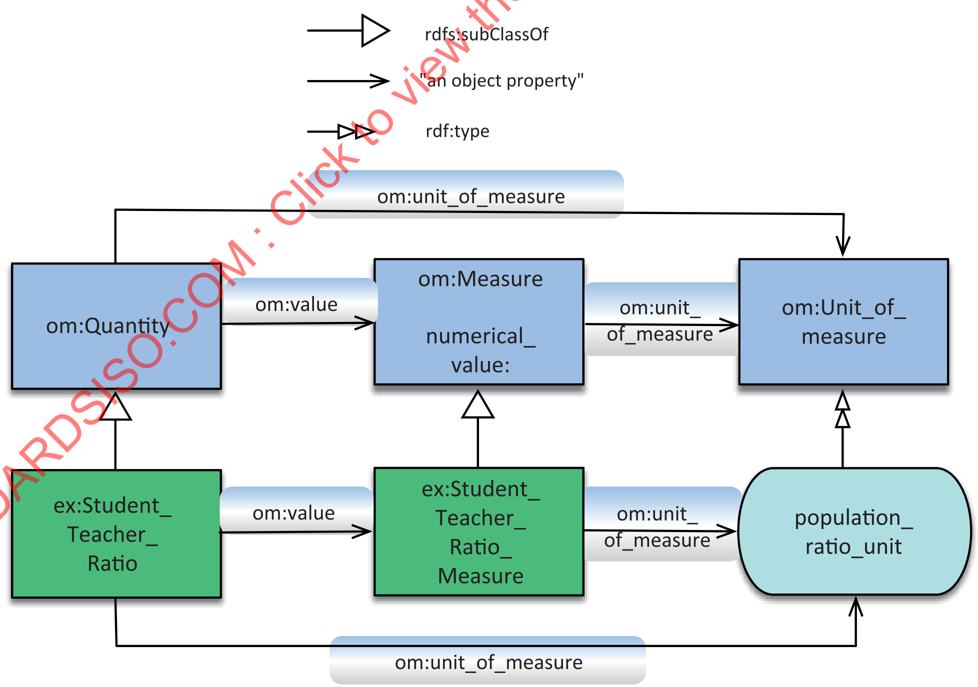


Figure 5 — Basic measurement classes

The concept of a `Quantity` is common across many standards. The ISO/IEC Guide 99:2007 defines a quantity as a "property of a phenomenon, body, or substance, where the property has a magnitude that can be expressed as a number and a reference". W3C defines a Quantity in www.w3.org/2007/ont/unit as a "(scalar) physical quantity or dimensionless number". QUDT (http://qudt.org/schema/quantity) defines a Quantity to be "the measurement of an observable property of a particular object, event, or physical system." The definition of Quantity adopted in this document is the OM version defined above (see Reference [31]).

Unit_of_measure is divided into three subclasses (Figure 6):

— Singular_unit, such as a metre;

— Unit_multiple_or_submultiple, defines multiples or submultiples of a Singular_unit. For example, if the singular unit is a metre, then a kilometre would be a multiple, and a centimetre would be a submultiple, there are other possible multiples and submultiples. A Unit_multiple_or_submultiple links to the singular unit it is a multiple of via the singular_unit property;

— Compound_unit denotes a combination of Unit_of_measure's. For example, speed would be an instance of a Unit_division where singular (e.g., metre) or multiple unit (e.g., kilometre) would be divided by time (e.g., hour).

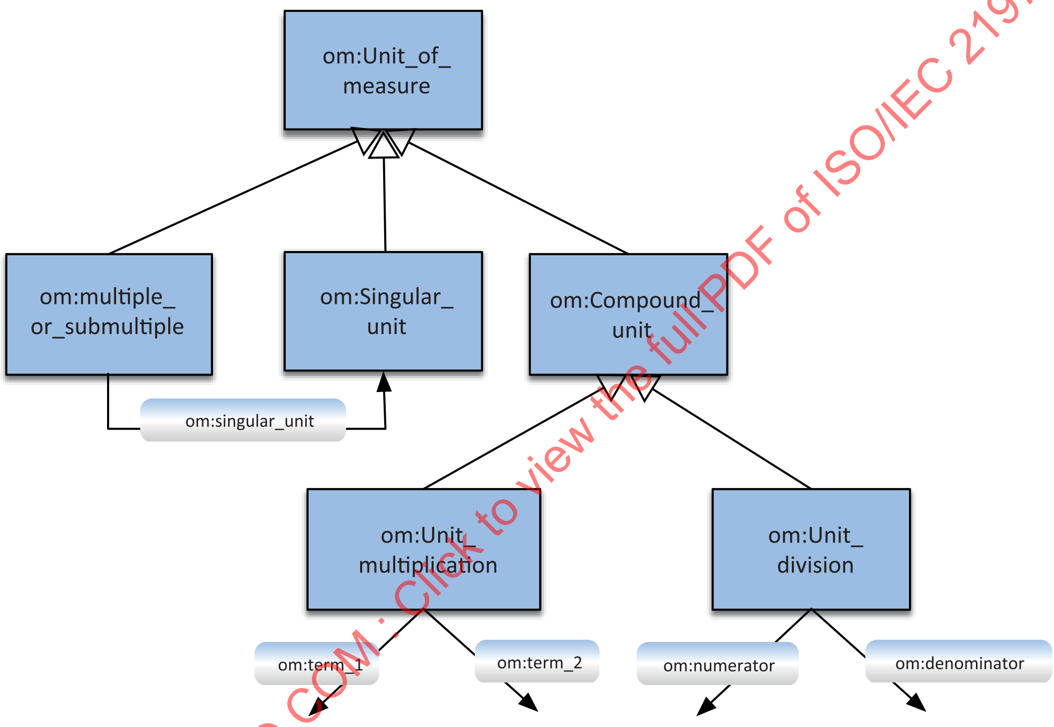


Figure 6 — Unit of measure taxonomy

Defining a unit of measure not only requires the specification of whether it is singular or compound, but whether the scale of the unit is nominal, ordinal, interval or ratio. The latter two scales are also called cardinal scales. An example of a scale is the Celsius scale, a temperature scale. For ratio scales, a zero point can be defined (Figure 7).

NOTE Ratio data on the ratio scale has measurable intervals. For example, the difference between a length of ten metres and nine metres is the same as the interval between two metres and three metres. Ratio scale scales have a meaningful zero whereas interval scales do not. The zero in a ratio scale means that something doesn't exist. For example, the zero in the Kelvin temperature scale means that heat does not exist.

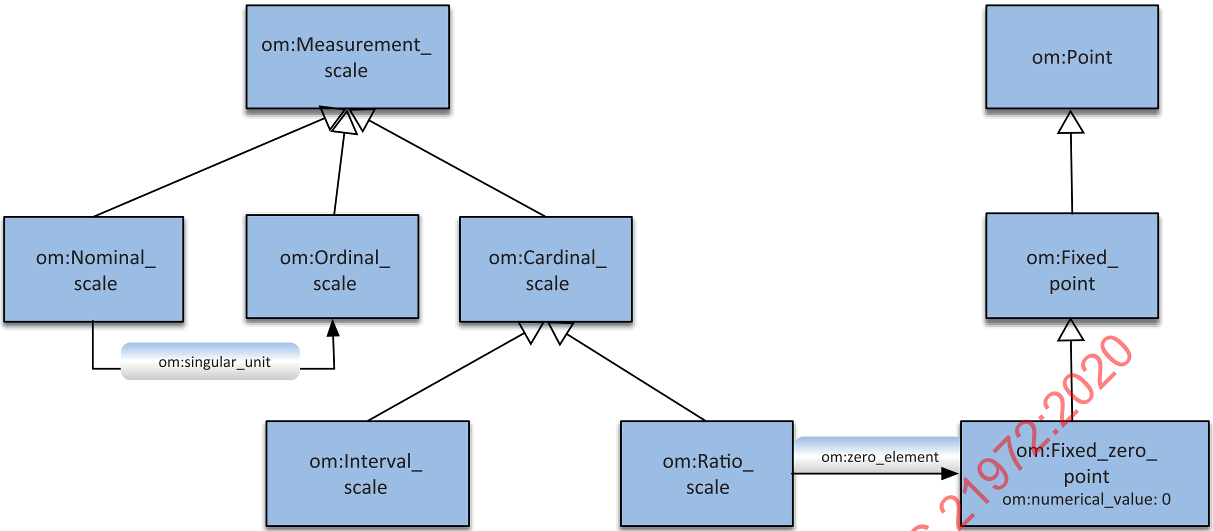


Figure 7 — Measurement scale taxonomy

7.3 Formal specification

Table 1 formally defines each of the classes of the measurement portion of this document.

Table 1 — Measurement Classes

Class	Property	Value Restriction
Quantity	value	exactly 1 Measure
	unit_of_measure	exactly 1 Unit_of_measure
	phenomenon	exactly 1 owl:Thing
Measure	unit_of_measure	exactly 1 Unit_of_measure
	numerical_value	exactly 1 xsd:string
Unit_of_measure	rdfs:subClassOf	owl:Thing
Singular_unit	rdfs:subClassOf	Unit_of_measure
Unit_multiple_or_submultiple	rdfs:subClassOf	Unit_of_measure
	prefix	exactly 1 Prefix
	singular_unit	exactly 1 Singular_unit
	symbol	min 1 xsd:String
Compound_unit	rdfs:subClassOf	Unit_of_measure
Unit_multiplication	rdfs:subClassOf	Compound_unit
	term_1	exatly 1 Unit_of_measure
	term_2	exactly 1 Unit_of_measure
Unit_division	rdfs:subClassOf	Compound_unit
	numerator	exactly 1 Unit_of_measure
	denominator	exactly 1Unit_of_measure
Measurement_scale	rdfs:subClassOf	rdfs:Class
Nominal_scale	rdfs:subClassOf	Measurement_scale
Ordinal_scale	rdfs:subClassOf	Measurement_scale
Cardinal_scale	rdfs:subClassOf	Measurement_scale
Interval_scale	rdfs:subClassOf	Cardinal_scale

Table 1 *(continued)*

Class	Property	Value Restriction
Ratio_scale	rdfs:subClassOf	Cardinal_scale
	zero_element	exactly 1 Fixed_zero_point
Fixed_zero_point	rdfs:subClassOf	Fixed_point
	numerical_value	value "0"
Fixed_point	rdfs:subClassOf	Point
Point	rdfs:comment	"A point is an element of an interval scale or a ratio scale, for example, 273,16 on the Kelvin scale indicates the triple point of water thermodynamic temperature." (OM, see Reference [31]).

8 Indicator quantities and units of measure

8.1 Core classes and properties

An indicator is a quantity that is a ratio of a numerator and a denominator that are also quantities. It has a city and time period associated with it. The numerator and denominator quantities can have different units of measure. One example of a unit of measure is the size of a population. A population_cardinality_unit is defined to be an individual of a Cardinality_unit that is a subclass of a Singular_unit. Figure 8 depicts the specification of the Cardinality_unit.

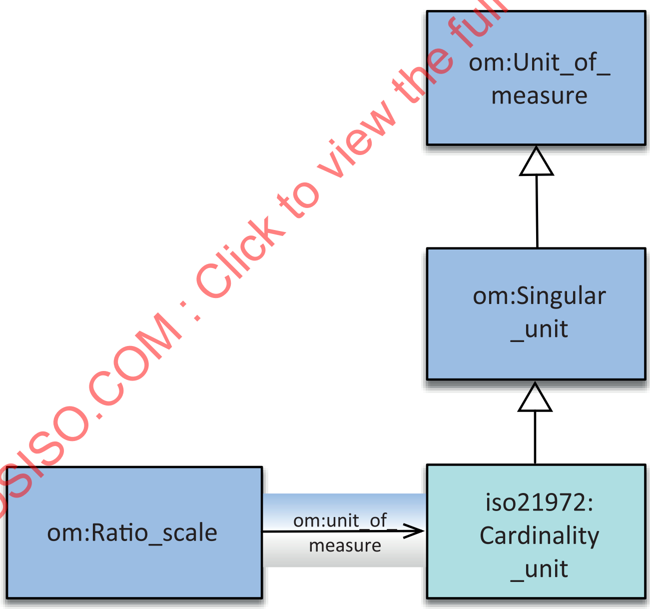


Figure 8 — Cardinality_unit definition

In Figure 9, population_cardinality_unit is depicted to be an instance of Cardinality_unit, which is the unit of measure for the cardinality of a set defined by a Population (defined in the next clause), and is associated with the symbol "pc". For example, 1 100 pc represents a population cardinality (or size) of 1 100. This document takes advantage of prefix notations to scale the numbers by defining units of measures: kilopc, megapc and gigapc, which are multiples of population_cardinality_unit. 1,1 kilopc represents 1 100 pc.

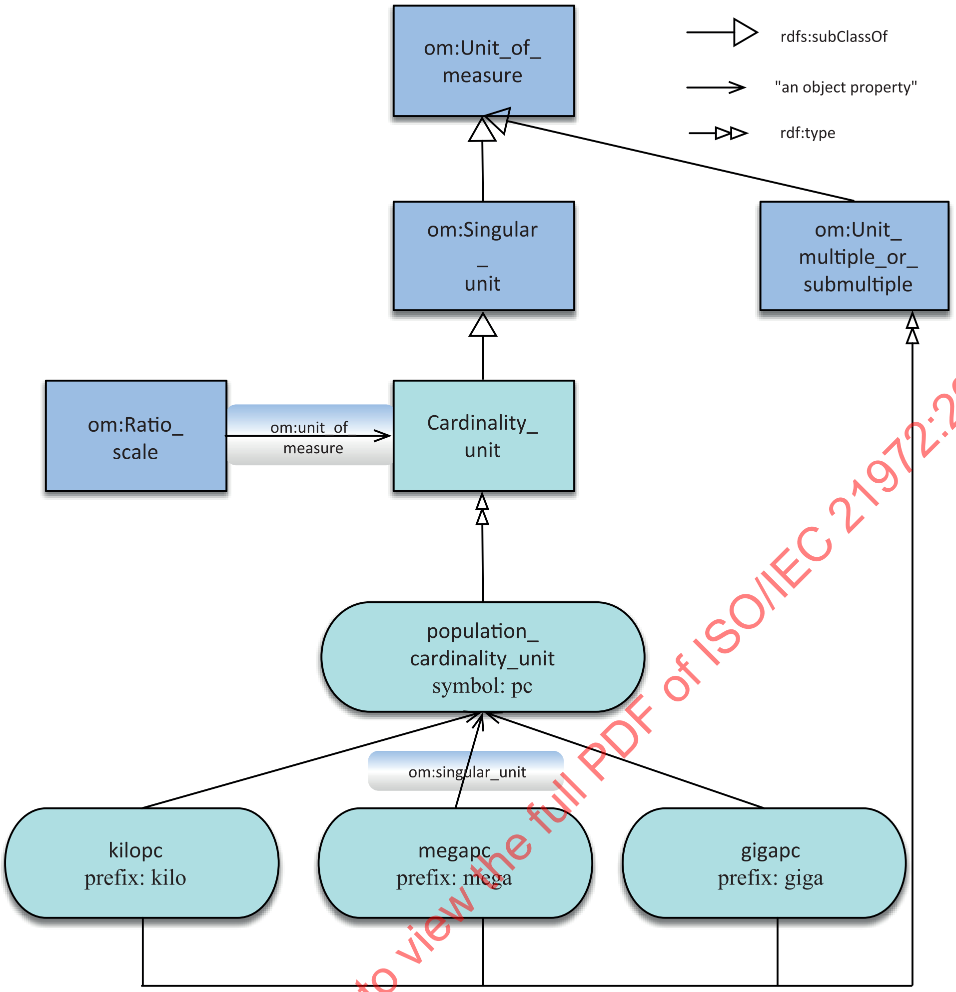


Figure 9 — Population_cardinality_unit definition

With the above defined, it is possible to introduce the unit of measure for measuring a population ratio. population_ratio_unit is defined to be an instance of Unit_division. It has two properties:

— numerator, whose range is restricted to being a population_cardinality_unit;

— denominator, whose range is restricted to being a population_cardinality_unit.

In other words, a population ratio is the ratio of two population cardinalities (i.e., number of members/ elements in each population).

Figure 10 depicts the definition of the population_ratio_unit.

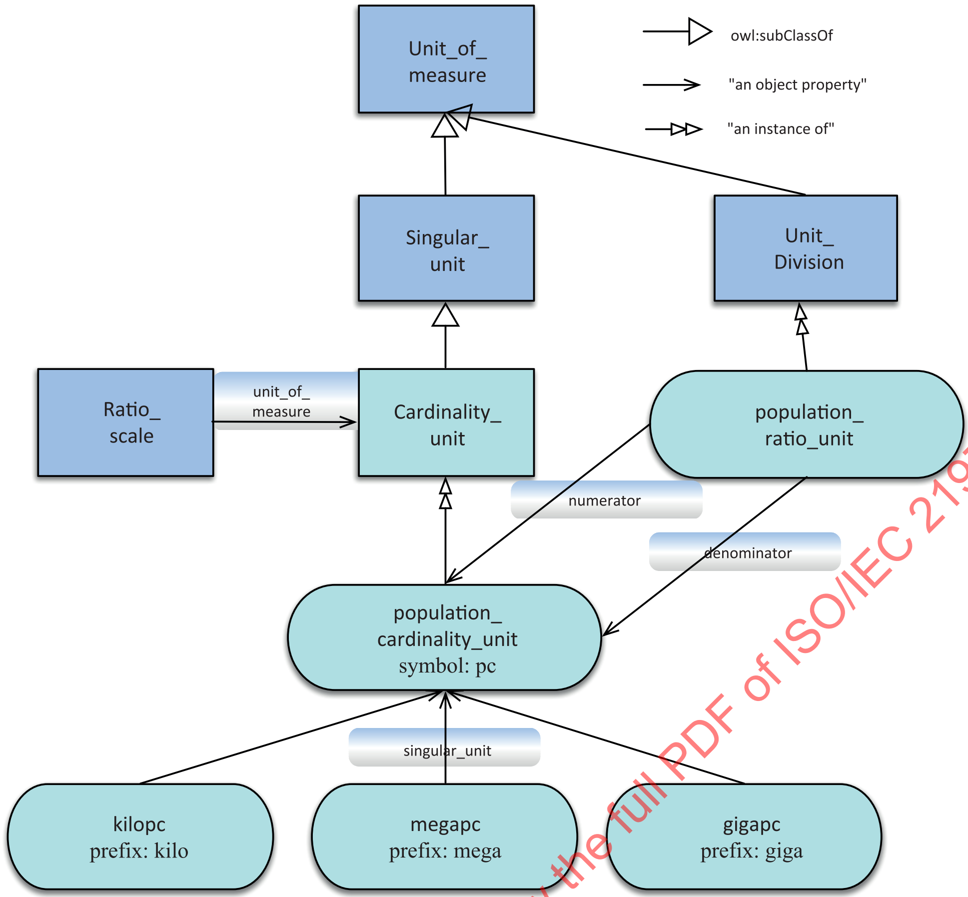


Figure 10 — (1 of 3)

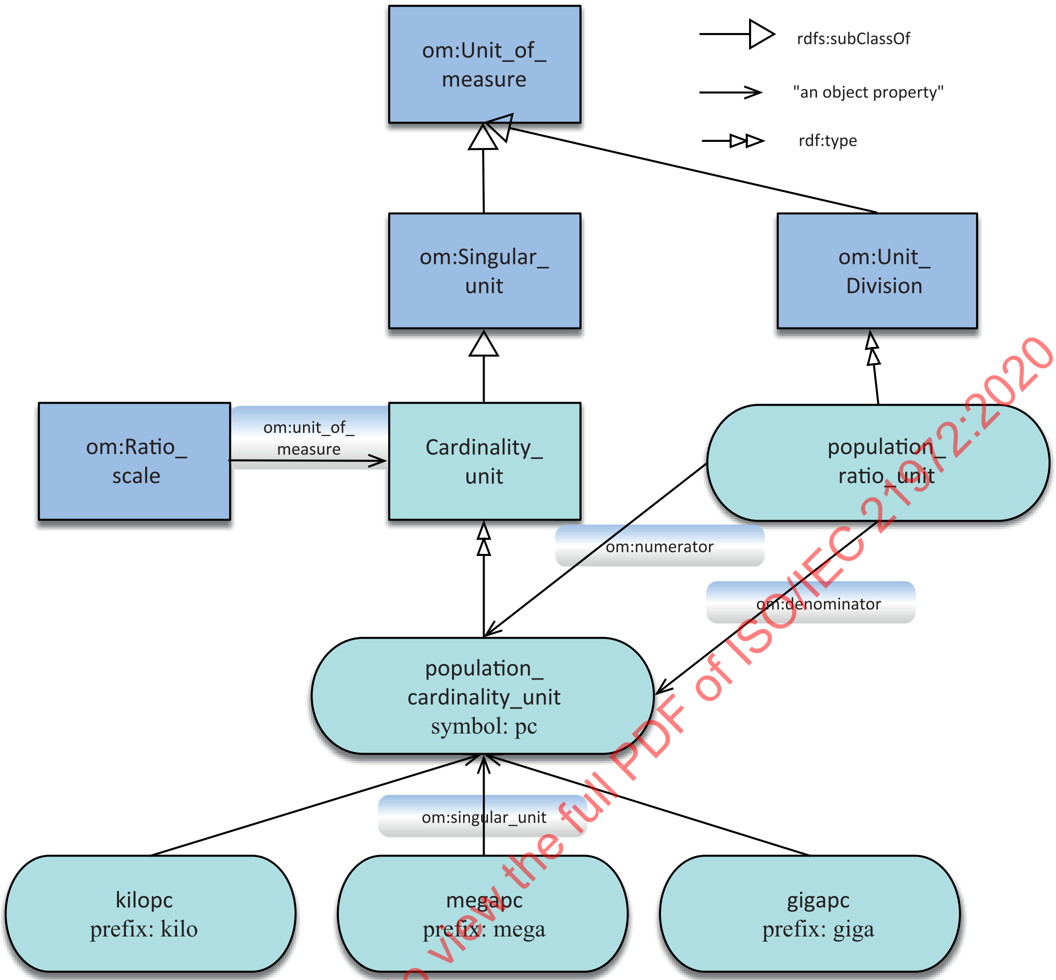


Figure 10 — (2 of 3)

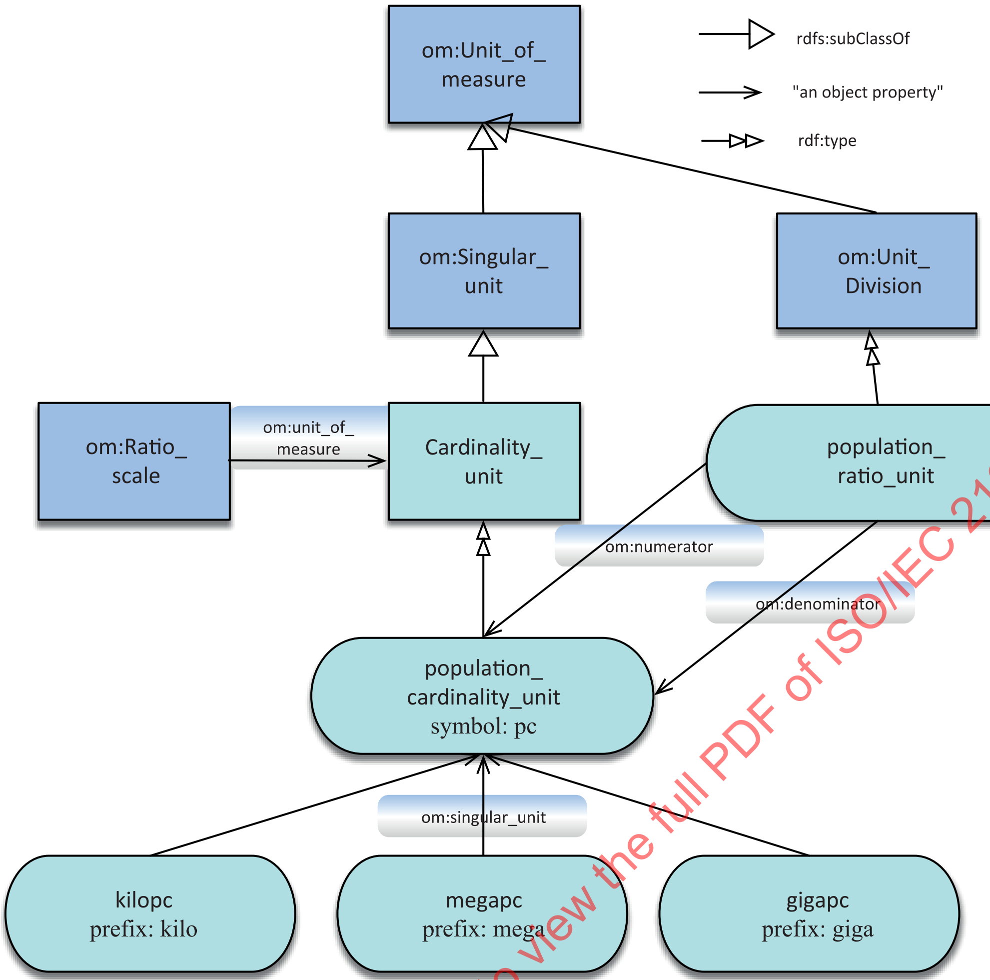


Figure 10 — Population_ratio_unit definition (3 of 3)

The above provides the unit of measures for populations (pc) and population ratios (pc/pc).

8.2 Formal specification

Table 2 specifies an indicator.

Table 2 — Indicator Classes

Class	Property	Value
Indicator	rdfs:subClassOf	Quantity
	unit_of_measure	exactly 1 Unit_of_measure
	value	exactly 1 Measure
	for_city	exactly 1 City
	for_time_interval	exactly 1 DateTImeInterval
RatioIndicator	rdfs:subClassOf	Indicator
	unit_of_measure	exactly 1 Unit_division
	numerator	exactly 1 Quantity
	denominator	exactly 1 Quantity
City	rdfs:subClassOf	sc:City

The specifications for population cardinality are given in Table 3.

Table 3 — Cardinality and Monetary Classes

Class	Property	Value
Cardinality_unit	rdfs:subClassOf	Singular_unit
	inverse unit_of_measure	exactly 1 Cardinality_scale
Cardinality_scale	rdfs:subClassOf	Ratio_scale
	zero_element	value fixed_zero_cardinality
fixed_zero_cardinality	rdf:type	Fixed_zero_point
	numerical_value	0
population_cardinality_unit	rdf:type	Cardinality_unit
	symbol	“pc”
Monetary_unit	rdfs:subClassOf	Singular_unit

With the definition of a population_cardinality_unit, the different types of singular units of measures and the compound units of measures upon which they are based on are specified in Table 4. The names of individuals of Monetary_unit shall adopt the codes for currencies in accordance with ISO 4217. For Unit_multiple_or_submultiple individuals prefix names, the ISO 80000 series shall be used.

Table 4 — Unit Individuals

Individual	Property	Value
decibel	rdf:type	Singular_unit
interruption	rdf:type	Singular_unit
occurrence	rdf:type	Singular_unit
gram	rdf:type	Singular_unit
microgram	rdf:type	Unit_multiple_or_submultiple
	prefix	micro
	singular_unit	gram
hour	rdf:type	Singular_unit
watt	rdf:type	Singular_unit
year	rdf:type	Singular_unit
minute_time	rdf:type	Singular_unit
square metre	rdf:type	Unit_exponentiation
	exponent	2
	base	metre
kilopc (represents 1 000 pc – population cardinality units)	rdf:type	Unit_multiple_or_submultiple
	prefix	kilo
	singular_unit	population_cardinality_unit
	symbol	“kilopc”
megapc (represents 1 000 000 pc – population cardinality units)	rdf:type	Unit_multiple_or_submultiple
	prefix	mega
	singular_unit	population_cardinality_unit
	symbol	“megapc”
gigapc (represents 10^9 pc – population cardinality units)	rdf:type	Unit_multiple_or_submultiple
	prefix	giga
	singular_unit	population_cardinality_unit
	symbol	“gigapc”

Table 4 *(continued)*

Individual	Property	Value
hecto_kilo_pc (represents 100 000 pc – population cardinality units)	rdf:type	Unit_multiple_or_submultiple
	singular_unit	population_cardinality_unit
	prefix	hecto_kilo
	symbol	"hecto_kilo_pc"
hecto_kilo (represents 100 000)	rdf:type	SI_prefix
USD	rdf:type	Monetary_unit
EUR	rdf:type	Monetary_unit

With the basic units defined, units of measure are defined in Table 5 for the various types of indicator ratios.

Table 5 — Indicator ratio units of measures

Individual	Property	Value
hour_per_interruption	rdf:type	Unit_division
	numerator	hour
	denominator	interruption
interruption_per_year	rdf:type	Unit_division
	numerator	interruption
	denominator	year
kwh_per_square_metre	rdf:type	Unit_division
	numerator	kilowatt_hour
	denominator	square_metre
kwh_per_pc	rdf:type	Unit_division
	numerator	kilowatt_hour
	denominator	population_cardinality_unit
kwh_per_year	rdf:type	Unit_division
	numerator	kilowatt_hour
	denominator	year
microgram_per_cubic_metre	rdf:type	Unit_division
	numerator	microgram
	denominator	cubic_metre
minute-time_per_occurrence	rdf:type	Unit_division
	numerator	minute-time
	denominator	occurrence
occurrence_per_hecto_kilo_pc	rdf:type	Unit_division
	numerator	occurrence
	denominator	hector_kilo_pc
pc_per_hecto_kilo_pc	rdf:type	Unit_division
	numerator	population_cardinality_unit
	denominator	hector_kilo_pc
population_ratio_unit	rdf:type	Unit_division
	numerator	population_cardinality_unit
	denominator	population_cardinality_unit

9 Statistics

9.1 General

An example of an indicator is student/teacher ratio (STR), which is based on a measure of the number of students and teachers within a population designated by a city. Both can be viewed as a statistical measurement in the sense that there is a population to be measured, namely a city population, and the number of members is counted that satisfy a description of a student and a teacher, respectively. While the STR requires a count of the population, other measures would require statistical measures of mean, deviation, etc. of other characteristics of the population.

There exist vocabularies/ontologies for the representation and publishing of statistical datasets. The Statistical Data and Metadata eXchange system(SDMX)[14] uses the Resource Description Framework (RDF)[16] triples to represent the observations that comprise the dataset. Its representation includes a mapping from a 'ComponentProperty' to a Class, thereby enabling a semantic interpretation of the property. Reference [30] defines a vocabulary for publishing census data. Their core class is an Observation that is linked to a dataset, geocode and value. The RDF Data Cube Vocabulary[15] provides a RDF representation for representing datasets. It provides terms for specifying the dimensions of a dataset, what is being measured and its units of measure, and individual observations. It uses the Dublin Core terms[17] for specifying meta-data for a dataset.

A different approach to a statistics ontology is the GovStat ontology[29]. GovStat provides the classes and properties for representing basic statistical concepts. The core class is the 'Population' to be measured. A 'Population' is linked to a 'Parameter' (e.g., mean, standard deviation) by the 'is_described_by' property. The GovStat ontology was selected as the basis for the Statistics ontology defined in this document.

9.2 Core concepts and properties

Figure 11 depicts a subset of the classes and properties of the GovStat ontology incorporated in this document. The core class is the Population to be measured. A Population is linked to a parameter (e.g., mean, standard deviation, cardinality) by the is_described_by property, and the parameter is a sub class of Parameter. Depending on the subclass of Parameter, there is a reverse link back to the Population.

- Cardinality parameter links back to the Population using cardinality_of.
- Sum parameter links back using the sum_of property.
- Mean parameter links back using the mean_of property.

Parameter links via the parameter_of_var property to a Variable which specifies the variable/data property that is to be summed, averaged, etc. across the members of the Population. Variable has a data property has_Name that specifies the variable/property over which the parameter is derived. It is specified as a string since a property cannot be a value of another property in Protégé's version of OWL (i.e., not full DL).

In statistics, it is almost always the case that only a portion of the population is measured. This portion is represented by the class Sample, and the parameter being measured is represented as a subclass of Statistic. Finally, the variable for which the parameter is being measured is defined by the class Observation which `Statistic` links to via the property is_composed_of, and the actual variable which is a subclass of Variable is linked to Observation via the property is_a_characteristic_of.

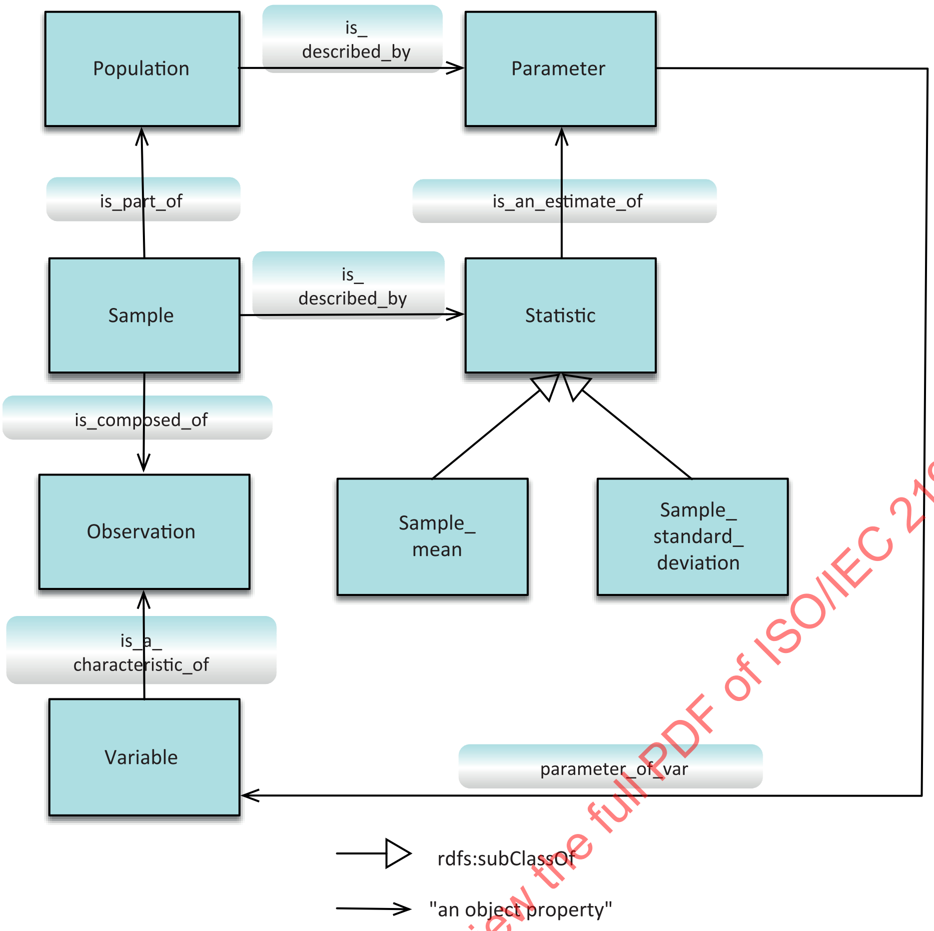


Figure 11 — Statistics ontology

9.3 Formal specification

Table 6 defines the basic statistics classes. The prefix "geo" refers to the Geonames namespace.

Table 6 — Statistics classes

Class	Property	Value
Population	defined_by	only owl:Thing
	located_in	exactly 1 Feature
	for_time_interval	only DateTimeInterval
	is_described_by	only Parameter
Feature	rdfs:subClassOf	geo:Feature
	rdfs:subClassOf	gis:Polygon
Parameter	rdfs:subClassOf	Quantity
	parameter_of_var	exactly 1 Variable
Variable	is_characteristic_of	only Observation
	has_Name	max 1 xsd:string
Sample	is_part_of	exactly 1 Population
	is_composed_of	only Observation
	is_described_by	only Statistic

Table 6 *(continued)*

Class	Property	Value
Observation	rdfs:subClassOf	owl:Thing
	member_of	only Population
	has_characteristic	exactly 1 Variable
Statistic	is_estimate_of	exactly 1 Parameter
Sample_mean	rdfs:subClassOf	Statistic
Sample_standard_deviation	rdfs:subClassOf	Statistic

Although properties are introduced as part of the concept definitions, Table 7 lists them separately.

Table 7 — Statistics properties

Property	Domain	Range
is_a_characteristic_of	Variable	Observation
has_characteristic	Observation	Variable
is_an_estimate_of	Statistic	Parameter
is_composed_of	(Sample or Population)	Observation
is_described_by	Population or Sample	Parameter or Statistic
is_part_of	Sample	Population
parameter_of_var	Parameter	Variable
member_of	Observation	Population

10 Populations

10.1 General

Measuring parameters of populations lies at the heart of city indicators. Populations do not refer only to people, but in a statistical sense to a finite collection of “things” under consideration. For education indicators, the populations include students and teachers. For environment indicators, the populations include sensor readings taken at different points in time. For fire and safety indicators, the populations include emergency call events. The following deconstructs the semantics of populations so that the definition of an indicator and its supporting data can be precisely modelled.

10.2 Core concepts and properties

10.2.1 Membership extent

The first question is what defines the members of a population? Consider the population of teachers. The ISO 37120:2018 definition states:

> “The number of classroom teachers and other instructional staff (e.g. teachers’ aides, guidance counselors) shall not include administrators or other non-teaching staff. Kindergarten or pre-school teachers and staff shall not be included.”

This definition assumes that the reader understands the terms teacher, classroom, instructional staff, guidance counselors, etc. In other words, it assumes that the reader has a “common sense” understanding of the domain of education. Hence, to model this population, the concept of Population has to be linked to the concepts contained in an education ontology (or at least the portions of the education domain relevant to the ISO 37120 Education theme) that define its members.

To accommodate the definition of a Population's member, the following property depicted in Figure 12 is added:

— **defined_by:** this property links a Population to a class that defines the necessary and/or sufficient conditions to be a member of the population.

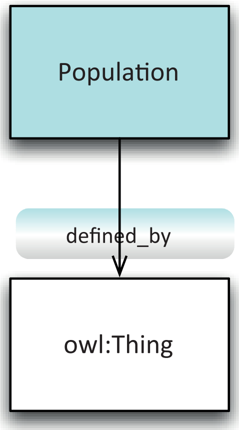


Figure 12 — Population member definition

Continuing the education example, to define a population of students, the Population class is linked to the gcie:Student class using defined_by. The definition of gcie:Student then has to be rich enough, as specified by the ISO 37120 Education theme, to determine membership in the Population. Figure 13 shows a portion of the numerator of the indicator pattern (Figure 2) specialized for the Student population. The Student population is defined_by the gcie:Student class. The Description Logic definition of the Student class provides the necessary and sufficient conditions for membership in the Student population. For example, one of those conditions is that it has a has_enrollment property linking to an Enrollment class. Clause 11 provides a detailed definition of the gcie:Student class.

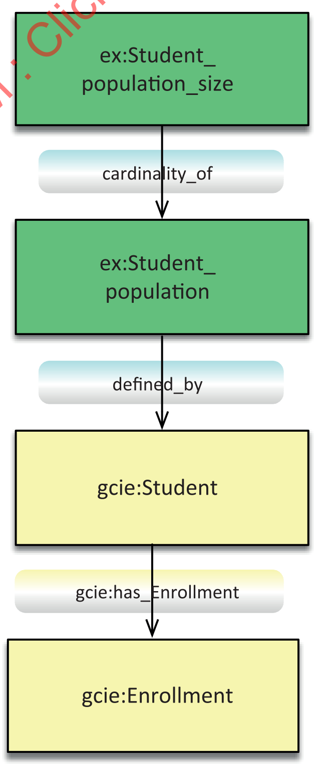


Figure 13 — Numerator of student/teacher ratio

10.2.2 Spatial extent

A second requirement for the representation of indicator populations is the specification of the spatial area from which the population is drawn. In theory, it is an entire city, but in practice it may be drawn from a larger or smaller area. Changes in spatial extent can be due to administrative changes such as amalgamation of cities, villages, etc., splitting of cities into more than one city, or can be due to a decision to not measure the city's entire population, but focus on a sub-area within the city. Whenever the spatial extent of a city changes, a new individual has to be created to represent the new version of that city.

There are two approaches to identifying a population's spatial extent. The first identifies a physical area by a place name (toponym), and the second by geo-spatial location. Toponyms can not only identify cities, but are used to identify any type of place, including villages, neighbourhoods, parks, buildings, hills, hospitals, and schools. But with the toponym approach, there is an additional requirement to *uniquely* identify a place. Names of cities are often duplicated throughout the world. How do people know that when people refer to Toronto, that it is the Toronto in Ontario, or the Toronto in Ohio or California? The semantic web provides a solution, namely each city or town of Toronto has an IRI that uniquely identifies it. For example, the UK's Ordnance Survey maintains a registry of placenames[28]. Geonames[32] is both a web site and a namespace. It contains over ten million place names, each having a unique IRI. For example, the geoname International Resource Identifier (IRI) for the City of Toronto, Ontario is http://www.geonames.org/6167865. The Ordnance Survey registry IRIs are linked to the Geonames using the OWL sameAs property, enabling the use of multiple IRIs to refer to the same place.

The root class of the Geonames ontology is the Feature. It contains a number of properties, including:

— **featureCode**: identifies the types of place, including:

  — A: Country, state, region, ...;

  — H: Stream, lake, ...;

  — L: Parks, area, ...;

  — P: City, village, ...;

  — R: Road, railroad, ...;

  — S: Spot, building, farm, ...;

  — T: Mountain, hill, rock, ...;

  — U: Undersea; V: Forest, heath, ...

  For example, P.PPLA is the feature code for a "seat of a first order administrative division."

— **parentCountry:** identifies the code of the country where the place is located in, as given in ISO 3166 (all parts).

— **wgs84_pos:lat** and **wgs84_pos:long**: identifies the latitude and longitude of the place using the wgs84_pos properties.

In parallel with the development of geonames, the Schema.org taxonomy has been developed to tag web page content with associated classes and properties, thereby making it easier for search engines to index content. Schema.org provides a rich set of place related classes, e.g., City, State, Airport, Mountain, that can be used to extend geoname Features. This is accomplished by making a specific IRI for a place like Toronto an instance (type) of a schema.org class such as City.

The second method of identifying the spatial extent of a population is to specify its geospatial location. This can be accomplished by specifying a polygon whose vertices are specified as latitude and longitude. A number of geometry ontologies exist, including ISO 19107[6], that provide for the representation of geometric classes such as Point, Line, Circle, and Polygon, and relating them using properties such as intersects, overlaps, contains, and disjoint. Place names can be integrated with geospatial information by making a specific place name an instance (type) of Polygon. The place name can be associated with the geospatial polygon that bounds the city, and relate it to other place names using geometry properties to other place names.

Figure 14 depicts the integration of both place name and geo-spatial information with Population. The property located_in is added to Population to specify both the IRI of the place name and its spatial geometry.

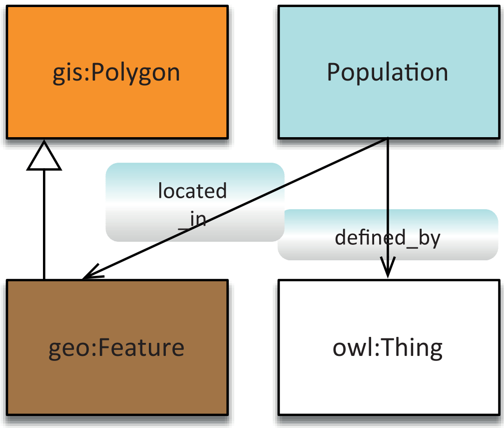


Figure 14 — Integration of spatial extent

10.2.3 Temporal extent

A third requirement for the representation of indicator populations is the specification of the time period over which the population is drawn. The periods over which populations are measured may vary by starting time, end time, duration and granularity. For example, in the student/teacher ratio indicator, the student population may have been drawn in 2012 and the denominator, teacher population, during 2013. Or more specific start and end times may be specified within the same year, such as February 1, 2013 to March 28, 2013 for the student population, and September 15 to November 30 for the teacher population. The ability to specify the time period at any level of granularity is important.

The OWL-Time ontology[26] satisfies these requirements. It provides for the representation of time points, intervals and durations, plus the specification of these at any level of granularity: year, month, week, day, hour, minutes and second. Secondly it provides a rich set of temporal relations, such as before, overlaps and during, that can be used to link and reason over the temporal points and intervals.

The definition of a Population is extended by adding the property for_time_interval. The range of the property is a DateTimeInterval that specifies the start and end date/times for the Population. Figure 15 depicts this addition.

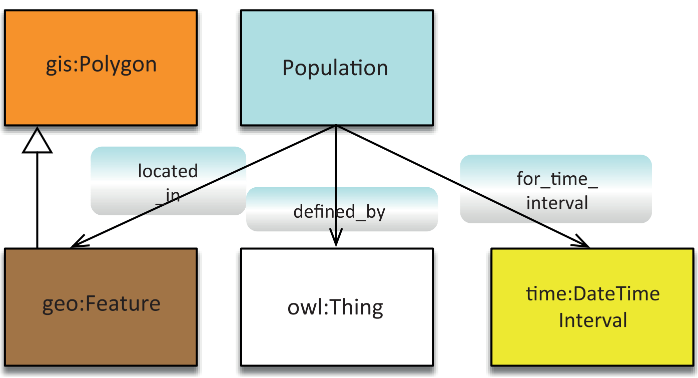


Figure 15 — Integration of temporal extent

10.2.4 Measured variable

The fourth requirement for populations is the representation of the parameter/statistic being measured. The student/teacher ratio indicator is based on a measure of the number of students and teachers within a population designated by a city (Placename). One can view both as a statistical measurement in the sense that there is a population to be measured, namely a city population, and the number of members is counted that satisfy a description of a student and a teacher, respectively. While the student/teacher ratio requires a count of the population, other indicators may require statistical measures of mean, deviation, etc. of other characteristics of the population.

Figure 16 depicts the extensions made to the statistics ontology to accommodate measuring the cardinality of the Population and the sum, mean and standard deviation of a property over the members of the Population. Sum, Mean, Standard_deviation and Cardinality are subclasses of Parameter. They have a reverse link back to Population, and a forward link, parameter_of_var, to the Variable they are measuring (except for Cardinality).

— Cardinality parameter links back to the Population using cardinality_of,

— Sum parameter links back using the sum_of property,

— Mean parameter links back using the mean_of property,

— Standard_deviation links back using the std_of property.

Variable contains the property has_Name that identifies the property contained in the class that defines membership in the Population, i.e., the value of the defined_by property. Due to limitations of DL, the value of has_Name is a string and not the actual property.

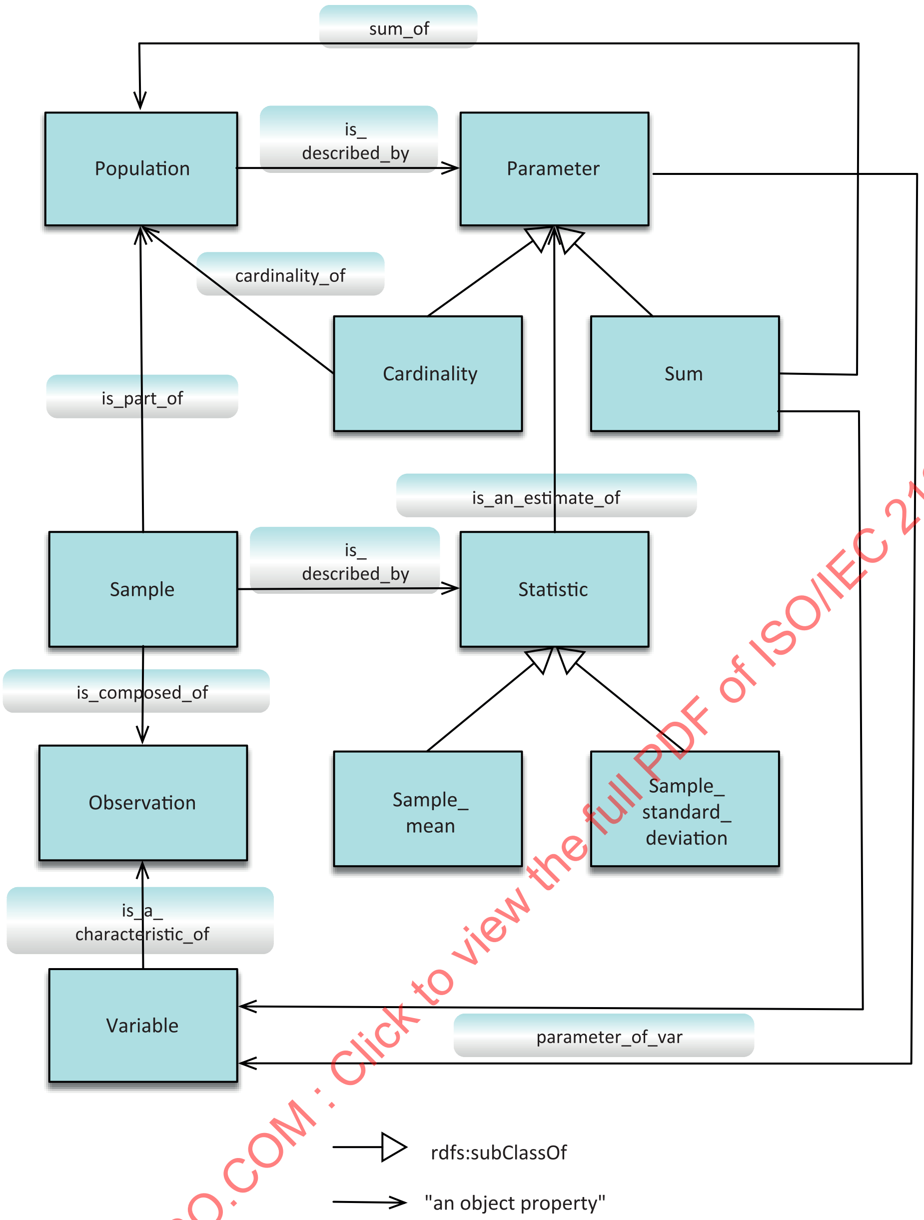


Figure 16 — GovStat extensions

Parameter links via the parameter_of_var property to a Variable which specifies the variable/data property that is to be summed, averaged, etc. across the members of the Population. Variable has a data property has_Name that specifies the variable/property over which the parameter is computed. It is specified as a string since a property cannot be a value of another property in Protégé's version of OWL (i.e., not full DL).

Each subclass of Parameter inherits the parameter_of_var property that links it to the variable (data property) of the members of the population for which the calculation is being performed (i.e., sum, mean, etc.).

Figure 17 depicts the integration of the Parameter Sum with Population, and the Variable over which the Sum is performed.

